



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/771,143	01/26/2001	Christopher Crim	CLARP027/P2616	6194
22434	7590	11/16/2004	EXAMINER	
BEYER WEAVER & THOMAS LLP P.O. BOX 778 BERKELEY, CA 94704-0778			PHAM, HUNG Q	
			ART UNIT	PAPER NUMBER
			2162	

DATE MAILED: 11/16/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

<b>Office Action Summary</b>	<b>Application No.</b>	<b>Applicant(s)</b>	
	09/771,143	CRIM ET AL.	
	<b>Examiner</b>	<b>Art Unit</b>	
	HUNG Q PHAM	2162	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

#### Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

#### Status

1) Responsive to communication(s) filed on 23 August 2004.

2a) This action is **FINAL**.                                    2b) This action is non-final.

3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

#### Disposition of Claims

4) Claim(s) 1,3,4,7,9-15,17-24,26,27 and 37 is/are pending in the application.

4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.

5) Claim(s) \_\_\_\_\_ is/are allowed.

6) Claim(s) 1,3,4,7,9-15,17-24,26,27 and 37 is/are rejected.

7) Claim(s) \_\_\_\_\_ is/are objected to.

8) Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

#### Application Papers

9) The specification is objected to by the Examiner.

10) The drawing(s) filed on \_\_\_\_\_ is/are: a) accepted or b) objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

#### Priority under 35 U.S.C. § 119

12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a) All    b) Some \* c) None of:

1. Certified copies of the priority documents have been received.

2. Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.

3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

#### Attachment(s)

1) Notice of References Cited (PTO-892) \*

2) Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)  
Paper No(s)/Mail Date \_\_\_\_\_.

4) Interview Summary (PTO-413)  
Paper No(s)/Mail Date. \_\_\_\_\_.

5) Notice of Informal Patent Application (PTO-152)

6) Other: \_\_\_\_\_.

**DETAILED ACTION*****Continued Examination Under 37 CFR 1.114***

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 08/23/2004 has been entered.

***Response to Arguments***

2. Applicant's arguments filed 08/23/2004 have been fully considered but they are not persuasive.

(a) As argued by applicant at page 8:

*However, it is respectfully submitted that a tree path for an object does NOT teach defining a calculation expression associated with at least one record in a database, wherein the calculation that can be evaluated at least partly based on at least one field of said at least one record.*

Examiner respectfully traverses because of the following reasons:

In order to control access to managed objects in a computer network, a database management system is used to store the management information in a set of database tables, and each database table stores the management information for corresponding managed objects in individual rows. Access to the management information is controlled by at least one permissions table. A

permissions table defines a subset of rows in the database tables that are accessible to at least one of the users (Col. 3, Lines 15-41). The tree path may be represented by FDN (Fully Distinguish Name), and the FDN operates as the primary key to the data stored in the table (Col. 19, Lines 35-36). Management information is requested by a user via SQL commands, for instance, to read the data in a table named "table 1" for a managed object whose FDN is equal to "/a/b/c," an authorized user named "Max" would use the SQL command: SELECT\*FROM view\_table1\_max WHERE FDN="a/b/c" (Col. 22-32). As seen, FDN is a record identifier or record ID.

The granted permission table is illustrated as below:

Granted Permissions Table for Table 1

1502	User Name	Object Name	Operation Type	1
	user_x	object_xyz	SELECT	
	user_x	object_qrs	UPDATE	
	user_y	object_xyz	SELECT	
	user_y	object_abc	DELETE	
	user_z	object_def	SELECT	
1510	group_a	object_hij	SELECT	
	group_z	object_jkl	SELECT	

A permission entry 1502 is tuple having three fields, user name, object name, and operation type. The object name, preferably, is the FDN or Full Distinguish Name (Col. 26, Lines 28-33). Referring to FIG. 11A as shown below, each row in the database tables includes a field called the Fully Distinguished Name or FDN of a managed object followed by columns of data. For example, an FDN can look like /systemid="sys1"/owner="accompany"/devicetype="router" (Col. 19, Lines 24-35).

Row	FDN	Data 1	...	Data N
-----	-----	--------	-----	--------

As seen, each row of the Granted Permissions Table is defined by a meaningful combination of a plurality of *expressions* to specify a record access right for a user, wherein *the expression* of column Object Name *associated with a record of the database* by the FDN. Each row in the Granted Permissions explicitly defines an access right of a user to a record in the database with its Fully Distinguished Name as a key is equal to the specified Fully Distinguished Name in the Granted Permissions Table. For example, based on the first row of the Granted Permissions Table, a User Name = user\_x has Operation Type = delete on any record that has Object Name = object\_xyz. As seen, each row expression in the Granted Permissions Table is *a calculation expression* with a plurality of implied EQUAL OPERATOR, and is evaluated by the FDN field of the record to determine the access right. In short, the Bapat technique as discussed performs the applicant's argument: *a calculation expression that can be evaluated at least partly based on at least one field of said at least one record.*

(b) As argued by applicant with respect to claim 3 at page 9, examiner respectfully traverses because of the following reasons:

Claim 3 recites *a method as recited in claim 1, wherein at least one expression is a calculation expression that can be evaluated at least partly based on at least one state variable or a field of a record of said database.* As discussed above, each row

in the Granted Permissions Table explicitly defines an access right of a user to a record in the database with its Fully Distinguished Name is equal to the specified Fully Distinguished Name in the Granted Permissions Table. For example, based on a row of the Granted Permissions Table, a user\_x can delete any record that has Object Name (FDN) = Record (FDN). As seen, each row expression in the Granted Permissions Table is a mathematical process evaluated by the FDN field of the record to determine the access right. In short, the Bapat technique as discussed performs the claimed *at least one expression is a calculation expression that can be evaluated at least partly based on at least one state variable or a field of a record of said database.*

(c) Applicant' argument with claims 17-18 and 23-26 have been considered but are moot in view of the new ground(s) of rejection.

### ***Claim Objections***

3. Claims 20 and 23 are objected to because of the following informalities: *a database system as recited in claim 39*, and claim 39 does not exist. Appropriate correction is required.

### ***Claim Rejections - 35 USC § 112***

4. The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make

and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

**5. Claim 1 and 37 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.**

Regarding claims 1 and 37, the claimed *calculated for each one of said plurality of users in order to determine whether each one of said plurality of users can access said at least one record* was not described in the specification.

#### ***Claim Rejections - 35 USC § 102***

**6. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:**

A person shall be entitled to a patent unless –

(e) the invention was described in a patent granted on an application for patent by another filed in the United States before the invention thereof by the applicant for patent, or on an international application by another who has fulfilled the requirements of paragraphs (1), (2), and (4) of section 371(c) of this title before the invention thereof by the applicant for patent.

The changes made to 35 U.S.C. 102(e) by the American Inventors Protection Act of 1999 (AIPA) and the Intellectual Property and High Technology Technical Amendments Act of 2002 do not apply when the reference is a U.S.

patent resulting directly or indirectly from an international application filed before November 29, 2000. Therefore, the prior art date of the reference is determined under 35 U.S.C. 102(e) prior to the amendment by the AIPA (pre-AIPA 35 U.S.C. 102(e)).

**7. Claims 1, 3-4, 7 and 9-10 are rejected under 35 U.S.C. 102(e) as being anticipated by Bapat et al. [USP 6,236,996 B1].**

Regarding claim 1, Bapat teaches a method and a computer program for controlling managed objects. Referring to FIG. 11A as illustrated below, each row in the database tables includes a field called the Fully Distinguished Name or FDN of a managed object followed by columns of data. For example, an FDN can look like /systemid="sys1"/owner="accompany"/devicetype="router" (Col. 19, Lines 24-35).

Row	FDN	Data 1	...	Data N

To limit user access to the management information stored in the tables, a database function called Views is used (Col. 19, Lines 55-57), created by create view procedure 362 (Col. 20, Lines 43-46), implemented by database access engine 286 of the DBMS 280 (Col. 19, Lines 57-58) via SQL commands (Col. 20, Lines 23-32) and view access controller 380 (Col. 22, Lines 28-30). As seen, the procedure, module and database access engine discussed above indicates a software that enable DBMS 280 to work with a user for accessing records stored

in database. In short, the technique as discussed performs the claimed *providing a database program as an interface for accessing records stored in said database*. As illustrated at FIG. 14, tables 310 and 320 as in FIG. 11A are stored in a conventional DBMS 280 (Col. 25, lines 49-50). Rows 311, 312, 321, 322 of the tables 310, 320 contain management information for managed objects (Col. 25, lines 60-61). The FDN operates as the primary key to the data stored in the table and to determine which managed objects that a particular user is permitted to access or modify (Col. 19, lines 36-40). Access control for a particular user on a particular managed object is defined by a permissions table as described below (Col. 26, lines 10-12).

Granted Permissions Table for Table 1

User Name	Object Name	Operation Type
user_x	object_xyz	SELECT
user_x	object_qrs	UPDATE
user_y	object_xyz	SELECT
user_y	object_abc	DELETE
user_z	object_def	SELECT
group_a	object_hij	SELECT
group_z	object_jkl	SELECT

1

A permission entry 1502 is tuple having three fields, user name, object name, and operation type. The object name, preferably, is the FDN or Full Distinguish Name for a managed object (Col. 26, Lines 28-33). The granted permission table is created by Create View procedure (Col. 20, Lines 63-67). As seen, each row of the Granted Permissions Table is defined by a meaningful combination of a plurality of *expressions* to specify a record access right for a user, wherein *the expression of column Object Name associated with a record of the database* by the FDN. Each row in the Granted Permissions explicitly defines an access right of a

user to a record in the database with its Fully Distinguished Name as a key is equal to the specified Fully Distinguished Name in the Granted Permissions Table. For example, based on the first row of the Granted Permissions Table, a User Name = user\_x has Operation Type = delete on any record that has Object Name = object\_xyz. As seen, each row expression in the Granted Permissions Table is a *calculation expression* with a plurality of implied EQUAL OPERATOR, and is evaluated by the FDN field of the record to determine the access right. In short, the Bapat technique as discussed performs the claimed *defining by said database program, at least one expression for at least one record of said database, wherein said at least one expression defines access right to said at least one record for a plurality of users of said database program, and wherein at least one expression is a calculation expression that can be calculated for each one of said plurality of users in order to determine whether each one of said plurality of users can access said at least one record*. When a user 300 issues an SQL command to access the DBMS 280 (Col. 22, lines 24-26, Col. 20, Lines 23-31) as the step of *receiving a request associated with a first user of said database program to access said at least one record*, view access controller 380 checks that the user has permission to query the View (Col. 22, Lines 28-30), and Access Control is enforced by *evaluating, by said database program, FDN of column Object Name as at least one calculation expression for said first user to determine whether said first user should be allowed access to said at least one record, and allowing said first user to access to said one record based on said evaluating of FDN as at least one expression for said first user* (Col. 27, line 45 to Col. 28, line 26).

Regarding claim 3, Bapat teaches all of the claimed subject matter as discussed above with respect to claim 1, Bapat further discloses *at least one expression is a calculation expression that can be evaluated at least partly based on at least one state variable or a field of record of said database* (FIG 11C, Col. 21, Lines 13-27).

Regarding claim 4, Bapat teaches all of the claimed subject matter as discussed above with respect to claim 1, Bapat further discloses *at least one expression can be defined based on fields and state variables of said database, and wherein said evaluating operates to return only one of two possible values, one of said possible values indicating that access to said at least one record should be granted, and the other one of said possible values indicating that access to said at least one record should be denied* (Col. 26, lines 28-33, Col. 27, line 45-Col. 28, line 26).

Regarding claim 7, Bapat teaches all of the claimed subject matter as discussed above with respect to claim 1, Bapat further discloses *defining of said at least one expression operates to define access privileges for a user of said database with respect to at least one operation that can be performed on one or more records of said database* (FIG. 15A and B).

Regarding claim 9, Bapat teaches all of the claimed subject matter as discussed above with respect to claim 1, Bapat further discloses *at least one user is assigned a password that is associated with said expression* (FIG. 15A and B).

Regarding claim 10, Bapat teaches all of the claimed subject matter as discussed above with respect to claim 1, Bapat further discloses *access to said at least one record can be for browsing, editing, or deleting of said at least one record* (FIG. 15A and B).

### ***Claim Rejections - 35 USC § 103***

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary.

Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of

35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

**9. Claims 11-15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Bapat et al. [USP 6,236,996 B1] in view of Elmasri et al. [Fundamentals of Database System].**

Regarding claim 11, Bapat teaches a method for controlling managed objects. As shown in FIG. 14, tables 310 and 320 as in FIG. 11A are stored in a conventional DBMS 280 (Col. 25, lines 49-50). Rows 311, 312, 321, 322 of the tables 310, 320 contain management information for managed objects (Col. 25, lines 60-61). The FDN operates as the primary key to the data stored in the table and to determine which managed objects that a particular user is permitted to access or modify (Col. 19, lines 36-40). Access control for a particular user on a particular managed object is defined by a permissions table as shown below (Col. 26, lines 10-12).

Granted Permissions Table for Table 1

User Name	Object Name	Operation Type
user_x	object_xyz	SELECT
user_x	object_qrs	UPDATE
user_y	object_xyz	SELECT
user_y	object_abc	DELETE
user_z	object_def	SELECT
group_a	object_hij	SELECT
group_z	object_jkl	SELECT

A permission entry 1502 is tuple having three fields, user name, object name, and operation type. The object name, preferably, is the FDN or Full

Distinguish Name for a managed object (Col. 26, Lines 28-33). Referring to FIG. 11A as shown below, each row in the database tables includes a field called the Fully Distinguished Name or FDN of a managed object followed by columns of data. For example, an FDN can look like

/systemid="sys1"/owner="accompany"/devicetype="router" (Col. 19, Lines 24-35).

Row	FDN	Data 1	...	Data N
-----	-----	--------	-----	--------

As seen, each row of the Granted Permissions Table is defined by a meaningful combination of characters or *expression* to specify a record access right for a user, wherein *the expression associated with a record of the database* by the FDN. Each row in the Granted Permissions explicitly defines an access right of a user to a record in the database with its Fully Distinguished Name as a key is equal to the specified Fully Distinguished Name in the Granted Permissions Table. For example, based on the first row of the Granted Permissions Table, a User Name = user\_x has Operation Type = delete on any record that has Object Name = object\_xyz. As seen, each row expression in the Granted Permissions Table is a *calculation expression* with a plurality of implied EQUAL OPERATOR, and is evaluated by the FDN field of the record to determine the access right. In short, the Bapat technique as discussed performs the claimed *defining at least one expression associated with at least one record of said database, wherein said at least one expression is a calculation expression that can be evaluated at least partly based on at least one field of said at least one record in said database, and wherein said*

*calculation expression defines access privileges of said one or more users with respect to at least one operation that may be requested to be performed by said one or more users on one or more records of said database.* When a user 300 issues an SQL command to access the DBMS 280 (Col. 22, lines 24-26, Col. 25, lines 65-67) for the status of all routers in the network or for information about a specified list of managed objects (Col. 28, lines 27-30) as *receiving a request to perform said at least one operation on one or more records of said database, said request being identified as a request made by said one or more users associated with user name.* Access Control is enforced by *evaluating user name, object name and operation type as said calculation expression for said one or more records, based on one or more fields of said one or more records, when said request has been received; said evaluation returning only one of two possible values, one of said possible values indicating that said at least one operation should be granted and another one of said possible values indicating that said at least one operation should be denied; granting said at least one operation to be performed when said evaluation returns one said possible value to indicate that said at least one operation should be granted; and denying said at least one operation to be performed when said evaluation returns one said another possible value to indicate that said at least one operation should be denied* (Col. 27, line 45-Col. 28, line 26). Elmasri teaches a method of protecting access to a database system by *identifying a password that is associated with one or more users of said database* (Elmasri, page 718). Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the

Bapat method by using a password to identify a user as taught by Elmasri in order to have a more secure database system.

Regarding claim 12, Bapat and Elmasri, in combination, teach all of the claimed subject matter as discussed above with respect to claim 11, Bapat further discloses *at least one operation can be a browse, an edit, or a delete operation* (FIG. 15A and B).

Regarding claim 13, Bapat and Elmasri, in combination, teach all of the claimed subject matter as discussed above with respect to claim 11, Bapat further discloses *calculation expression is not explicitly defined for said at least one operation but said calculation expression is one that has been defined for another operation which has been considered as a related operation to said at least one operation* (FIG. 15A).

Regarding claim 14, Bapat and Elmasri, in combination, teach all of the claimed subject matter as discussed above with respect to claim 11, Bapat further discloses *calculation expression can be evaluated at least partly based on a value of at least one field of said at least one record, and wherein said calculation expression can be evaluated at least partly based on at least one state variable of said database* (Col. 26, lines 28-33).

Regarding claim 15, Bapat and Elmasri, in combination, teach all of the claimed subject matter as discussed above with respect to claim 14, Bapat further discloses the step of *granting temporary or limited access to said at least one record to allow said evaluating of said calculation expression* (FIG. 15A).

**10. Claims 37 and 17-18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Bapat et al. [USP 6,236,996 B1] in view of Glasser et al. USP 6,308,173 B1].**

Regarding claim 37, Bapat teaches a method and a computer program for controlling managed objects. Referring to FIG. 11A as illustrated below, each row in the database tables includes a field called the Fully Distinguished Name or FDN of a managed object followed by columns of data. For example, an FDN can look like /systemid="sys1"/owner="accompany"/devicetype="router" (Bapat, Col. 19, Lines 24-35).

Row	FDN	Data 1	...	Data N

To limit user access to the management information stored in the tables, a database function called Views is used (Bapat, Col. 19, Lines 55-57), created by create view procedure 362 (Bapat, Col. 20, Lines 43-46), implemented by database access engine 286 of the DBMS 280 (Bapat, Col. 19, Lines 57-58) via SQL commands (Bapat, Col. 20, Lines 23-32) and view access controller 380 (Bapat, Col. 22, Lines 28-30). As seen, the procedure, module and database

access engine discussed above indicates a software that enable DBMS 280 to work with a user for accessing records stored in database. As illustrated at FIG. 14, tables 310 and 320 as in FIG. 11A are stored in a conventional DBMS 280 (Bapat, Col. 25, lines 49-50). Rows 311, 312, 321, 322 of the tables 310, 320 contain management information for managed objects (Bapat, Col. 25, lines 60-61). The FDN operates as the primary key to the data stored in the table and to determine which managed objects that a particular user is permitted to access or modify (Bapat, Col. 19, lines 36-40). Access control for a particular user on a particular managed object is defined by a permissions table as described below (Bapat, Col. 26, lines 10-12).

Granted Permissions Table for Table 1

User Name	Object Name	Operation Type
user_x	object_xyz	SELECT
user_x	object_qrs	UPDATE
user_y	object_xyz	SELECT
user_y	object_abc	DELETE
user_z	object_def	SELECT
group_a	object_hij	SELECT
group_z	object_jkl	SELECT

1

A permission entry 1502 is tuple having three fields, user name, object name, and operation type. The object name, preferably, is the FDN or Full Distinguish Name for a managed object (Bapat, Col. 26, Lines 28-33). The granted permission table is created by Create View procedure (Bapat, Col. 20, Lines 63-67). As seen, each row of the Granted Permissions Table is defined by a meaningful combination of a plurality of *expressions* to specify a record access right for a user, wherein *the expression* of column Object Name associated with a record of the database by the FDN. Each row in the Granted Permissions

explicitly defines an access right of a user to a record in the database with its Fully Distinguished Name as a key is equal to the specified Fully Distinguished Name in the Granted Permissions Table. For example, based on the first row of the Granted Permissions Table, a User Name = user\_x has Operation Type = delete on any record that has Object Name = object\_xyz. As seen, each row expression in the Granted Permissions Table is a *calculation expression* with a plurality of implied EQUAL OPERATOR, and is evaluated by the FDN field of the record to determine the access right. In short, the Bapat technique as discussed performs the claimed *defining at least one expression for at least one record of said database, wherein said at least one expression defines access right to said at least one record for a plurality of users of said database program, and wherein said at least one expression is a calculation expression that can be calculated for each one of said plurality of users in order to determine whether each one of said plurality of users can access said at least one record*. When a user 300 issues an SQL command to access the DBMS 280 (Bapat, Col. 22, lines 24-26, Col. 20, Lines 23-31) as the step of *receiving a request associated with a first user of said database program to access said at least one record*, view access controller 380 checks that the user has permission to query the View (Bapat, Col. 22, Lines 28-30), and Access Control is enforced by *evaluating, by said database program*, FDN of column Object Name as *at least one calculation expression for said first user to determine whether said first user should be allowed access to said at least one record*, and *allowing said first user to access to said one record based on said evaluating of FDN as at least one expression for said first user* (Bapat, Col. 27, line 45 to Col. 28, line 26). Bapat does not

explicitly teach the claimed *providing a Graphical User Interface which can be used to define expression*. However, as disclosed by Bapat, the system administrator 302 creates the permissions tables prior to use of the DBMS 280 by end users. The system administrator 302 invokes a call 440 to the *Create\_Permissions\_Tables* 442 procedure of the DBMS 280 (Bapat, Col. 26, lines 18-27). As seen, in order to create the permission table by the *Create\_Permissions\_Tables* procedure, obviously, a *Graphical User Interface* can be used to enter the user name, FDN and access control code as discussed above. Glasser teaches a Graphical User Interface for defining access control expression (Glasser, FIG. 6B). Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to include a Graphical User Interface as taught by Glasser in order to have a friendly system to define access right for a user.

Regarding claim 17, Bapat and Glasser, in combination, teach all of the claimed subject matter as discussed above with respect to claim 37, Glasser further discloses *Graphical User Interface operates to provide the ability for a user of said database to define an expression associated with at least one operation that may be requested to be performed by another user of said database on said one or more records stored in said database* (Glasser, FIG. 6B).

Regarding claim 18, Bapat and Glasser, in combination, teach all of the claimed subject matter as discussed above with respect to claim 37, Glasser

further discloses *Graphical User Interface operates to provide the ability for a user to define said expression without requiring said user to write a programming script* (Glasser, FIG. 6B).

**11. Claims 19-24 and 26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Bapat et al. [USP 6,236,996 B1] in view of Glasser et al. [USP 6,308,173 B1] and Elmasri et al. [Fundamentals of Database System].**

Regarding claim 19, Bapat and Glasser, in combination, teach all of the claimed subject matter as discussed above with respect to claim 37, but fail to teach *Graphical User Interface provides a window that allows a user to interact with said Graphical User Interface to identify a password for which access privileges may be defined or re-defined*. Elmasri teaches a method of protecting access to a database system by identifying a password that is associated with one or more users of said database (Elmasri, page 718). In the teaching of creating the permission table (Bapat, Col. 26, lines 16-50), a Graphical User Interface provides a window is implied. In FIG. 15A, a user name is identified by system administrator and the user access right is mapped to the table by *Create\_Permissions\_Tables* procedure. Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the combined Bapat and Glasser system by using a Graphical User

Interface to identify a password instead of user name in order to define access privilege for a user.

Regarding claim 20, Bapat, Glasser and Elmasri, in combination, teach all of the claimed subject matter as discussed above with respect to claim 19, Glasser further discloses *Graphical User Interface further provides a window that allows a user to specify a calculation expression which defines access privileges with respect to at least one operation that may be requested to be performed on said one or more records* (Glasser, FIG. 6B).

Regarding claim 21, Bapat, Glasser and Elmasri, in combination, teach all of the claimed subject matter as discussed above with respect to claim 20, Bapat further discloses *at least one operation can be a browse, edit, or a delete operation* (Bapat, FIG. 15A).

Regarding claim 22, Bapat, Glasser and Elmasri, in combination, teach all of the claimed subject matter as discussed above with respect to claim 20, Bapat further discloses *calculation expression can be evaluated at least partly based on a value in at least one field of said one or more records of said database, and wherein said calculation expression can be evaluated at least partly based on at least one state variable of said database* (Bapat, FIG 11C, Col. 21, Lines 13-27).

Regarding claim 23, Bapat, Glasser and Elmasri, in combination, teach all of the claimed subject matter as discussed above with respect to claim 19, Bapat further discloses *database program operates to determine whether access to at least one of said one or more records should be granted or denied* (Bapat, Col. 27, line 45-Col. 28, line 26).

Regarding claim 24, Bapat, Glasser and Elmasri, in combination, teach all of the claimed subject matter as discussed above with respect to claim 23, Bapat further discloses the step of *determining of whether access to said at least one record should be granted or denied is performed by evaluating a calculation expression for said at least one of said one record* (Bapat, Col. 27, line 45-Col. 28, line 26).

Regarding claim 26, Bapat, Glasser and Elmasri, in combination, teach all of the claimed subject matter as discussed above with respect to claim 24, Bapat further discloses *access to said at least one record can be for browsing, editing, or deleting of said record* (Bapat, FIG. 15A).

**12. Claim 27 is rejected under 35 U.S.C. 103(a) as being unpatentable over Bapat et al. [USP 6,236,996 B1] in view of Glasser et al. [USP 6,308,173 B1], Elmasri et al. [Fundamentals of Database System] and Gorelik et al. [USP 6,651,067 B1].**

Regarding claim 27, Bapat, Glasser and Elmasri, in combination, teach all of the claimed subject matter as discussed above with respect to claim 24, but fail to disclose *the database further comprises a cache, and wherein said cache operates to store an evaluated result of at least one calculation expression*. Gorelik teaches a database comprises a cache, and cache operates to store an evaluated result (Gorelik, FIG. 3). Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the combined Bapat, Glasser and Elmasri system by including a cache and store the result in cache as taught by Gorelik in order to speed up the retrieval process.

***Conclusion***

13. Any inquiry concerning this communication or earlier communications from the examiner should be directed to HUNG Q PHAM whose telephone number is 571-272-4040. The examiner can normally be reached on Monday-Friday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, JOHN E BREENE can be reached on 571-272-4107. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

14. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Examiner Hung Pham  
November 3, 2004



SHAHID ALAM  
PRIMARY EXAMINER